

Mobile Robot Navigation

Utilizing the Robot Operating System (ROS)

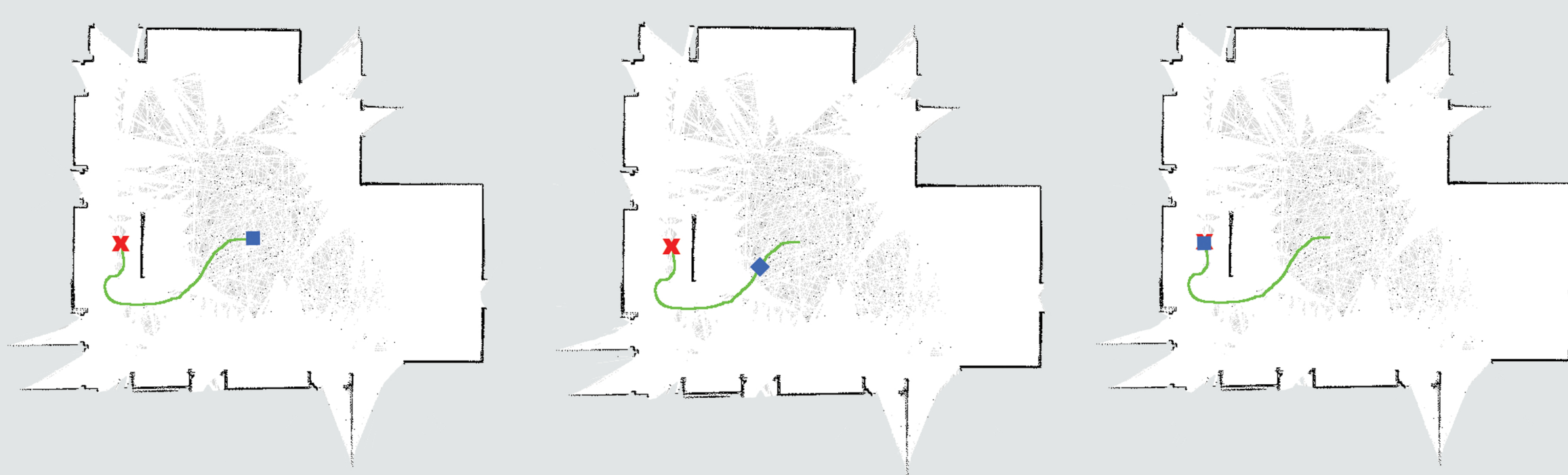


Bryan Custer : bat47@unh.edu Zach Piispanen : zpg3@unh.edu Jeremy Smith : jd58@unh.edu
Scott Kiesel : skiesel@cs.unh.edu Wheeler Ruml : ruml@cs.unh.edu

Objective

The goal of this project is to have a robot autonomously deliver small parcels from room to room on the 2nd floor of the UNH Engineering building, Kingsbury Hall.

The approach is to use ROS as a navigation platform and a web server for specifying goal locations. The P3-DX Pioneer will navigate through Kingsbury towards these goals. Below is a simple example using a partially complete map; the robot is specified by the blue box, the goal location as the red 'X', and the globally planned path as the green line.



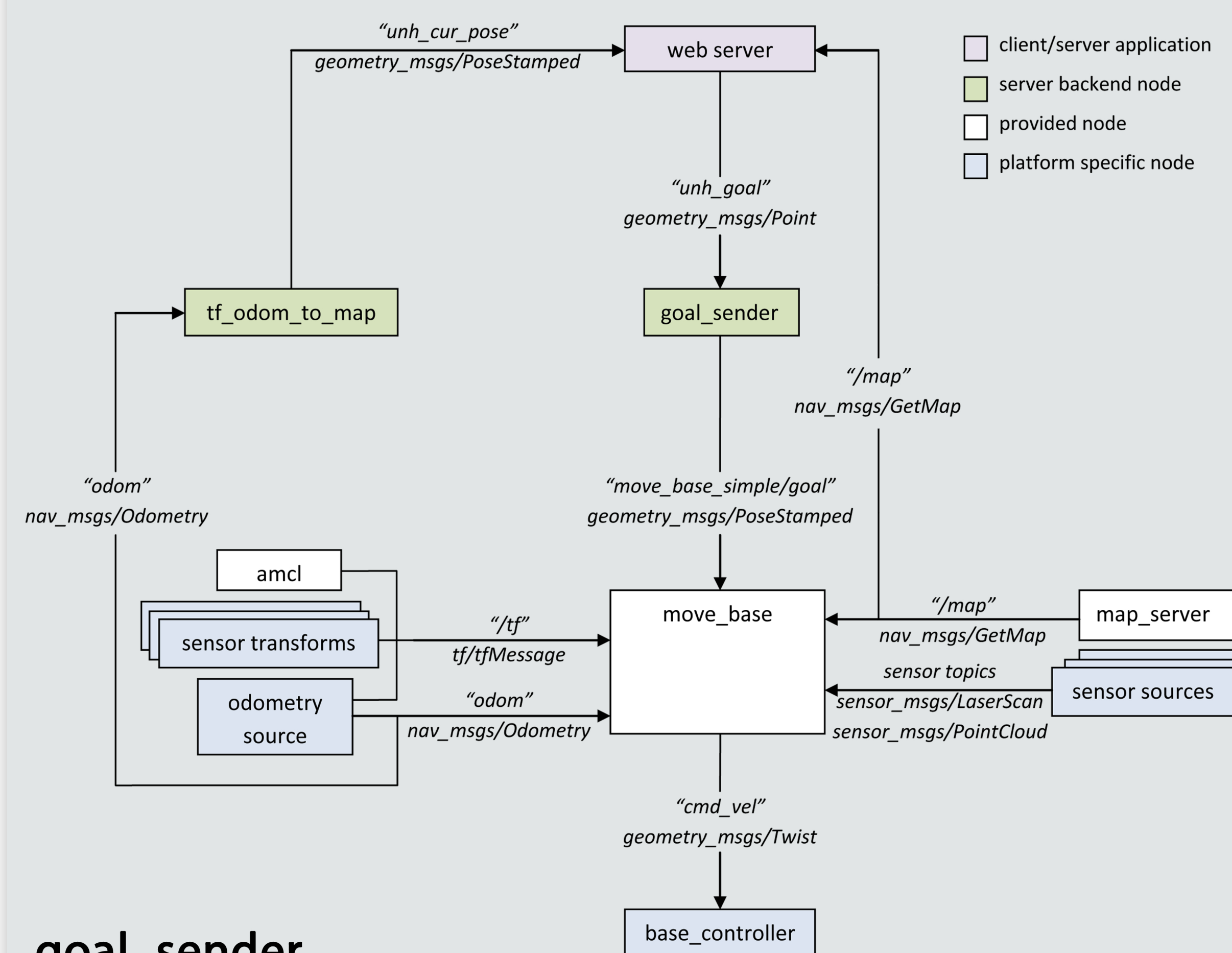
ROS

ROS is an open source software package that provides libraries comprised of stacks and packages (containing nodes) which allows users to build custom robotic applications. The navigation stack consists of general purpose navigation tools that act as building blocks for this project, such as SLAM, AMCL, Odometry sources, sensor transforms, and a map server.

SLAM (Simultaneous Localization And Mapping) is used to construct a map, and AMCL (Adaptive Monte Carlo Localization) is used to localize the robot within the given map. AMCL uses the laser range finder data to probabilistically determine its location via particle filter.

Node Overview

The diagram below shows how ROS nodes communicate with each other within the navigation stack. Custom nodes `tf_odom_to_map` and `goal_sender`, and the web server application were created specifically for this project and the rest were provided in the basic ROS navigation stack.

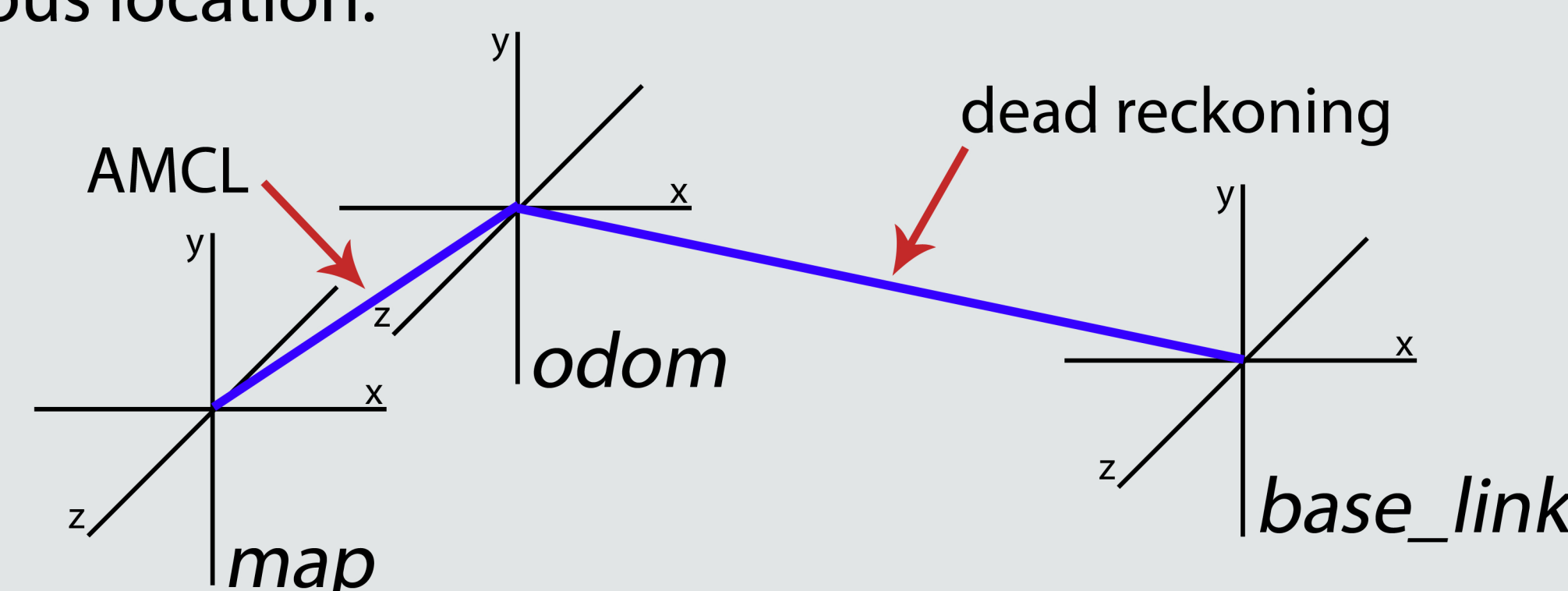


goal_sender

Points published by the web server are received by the `goal_sender`. Points come in as a tuple (x, y, z) where x and y are used to create a navigation goal within ROS; z is not used as the navigation of this project is 2D.

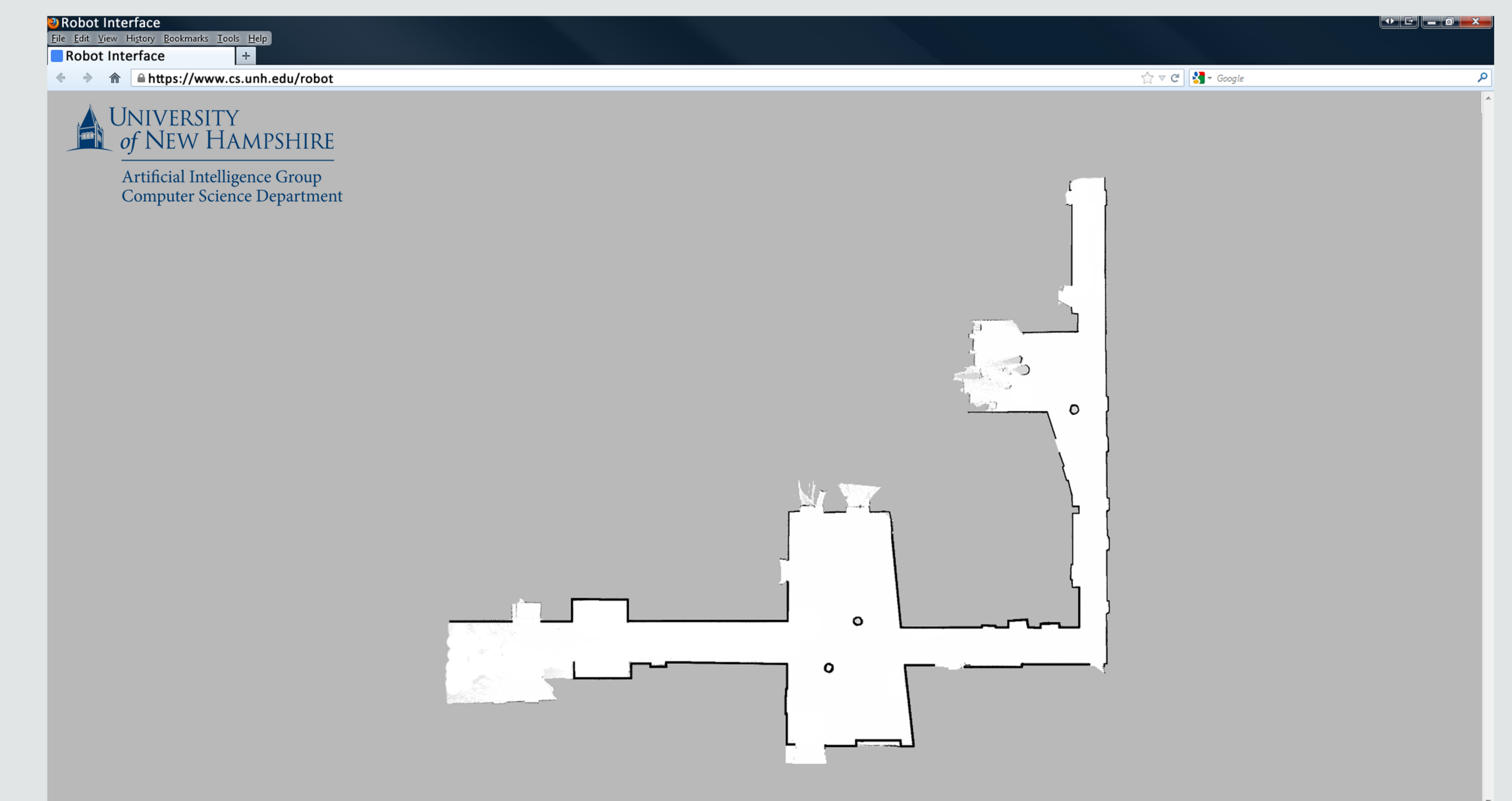
tf_odom_to_map

This node transforms a point, the robot's location, from the odom frame to the map frame. The odom frame contains information collected based on dead reckoning, or the estimated displacement based on movement from a previous location.



Web Interface

Users interact with the robot through clicking on the map image of Kingsbury Hall. A point is sent by JavaScript through ROS Bridge to the robot. It is published on a topic `unh_goal` which `goal_sender` subscribes to. In addition to publishing `unh_goal`, the web page subscribes to `unh_cur_pose` (the current location of the robot within the map) and `map` (the occupancy grid).



The web interface is a constantly evolving aspect of this project and this image shows just an early design of the visuals as we focused on the programming behind it.

Publisher-Subscriber Model

ROS utilizes a robust publisher-subscriber model.

